Recovering 3D Human Poses and Camera Motions from Deep Sequence

Takashi Shimizu, Fumihiko Sakaue and Jun Sato

Department of Computer Science and Engineering, Nagoya Institute of Technology, Gokiso, Showa, Nagoya 466-8555, Japan

Keywords: Human Poses, Camera Motions, CNN, RNN, LSTM, Deep Learning.

Abstract: In this paper, we propose a novel method for recovering 3D human poses and camera motions from sequential images by using CNN and LSTM. The human pose estimation from deep learning has been studied extensively in recent years. However, the existing methods aim to classify 2D human motions in images. Although some methods have been proposed for recovering 3D human poses recently, they only considered single frame poses, and sequential properties of human actions were not used efficiently. Furthermore, the existing methods recover only 3D poses relative to the viewpoints. In this paper, we propose a method for recovering 3D human poses and 3D camera motions simultaneously from sequential input images. In our network, CNN is combined with LSTM, so that the proposed network can learn sequential properties of 3D human poses and camera motions efficiently. The efficiency of the proposed method is evaluated by using real images as well as synthetic images.

1 INTRODUCTION

In recent years, human poses and actions are measured and used in various applications, such as movies and games. The motion capture systems are often used for measuring human poses and actions (Lab, 2003; Shotton et al., 2011). While the early motion capture systems (Lab, 2003) require special markers on the human body, recent systems such as Kinect sensors (Shotton et al., 2011) do not need to use markers. Although these motion capture systems are very useful for short range measurements in well-maintained environments, they cannot be used for long range measurements or uncontrolled environments, such as outdoor scenes. In such situations, passive methods such as camera based pose recognition methods are very useful.

For measuring human poses and actions from camera images, silhouette images were often used for neglecting the texture of clothes etc. (Agarwal and Triggs, 2004; Sminchisescu and Telea, 2002). The shading information was also used for estimating 3D poses from a single view (Guan et al., 2009). More recently, the deep learning has been used for pose estimation (Toshev and Szegedy, 2014). As shown in many recent papers, the deep learning provides us with the state of the art accuracy in various fields (LeCun et al., 1989; LeCun et al., 1998; Le et al., 2011; Le, 2013; Taylor et al., 2010), and the use of deep learning in the human action recognition is promising. Although many neural nets have been proposed for recognizing 2D human poses and actions (Toshev and Szegedy, 2014), the research on neural nets for 3D human pose recovery has just started (Chen and Ramanan, 2017; Tome et al., 2017; Lin et al., 2017; Mehta et al., 2017), and it requires more work to obtain better accuracy and to use in various situations. In particular, most of the current works on 3D human pose recovery are based on a single image (Chen and Ramanan, 2017; Tome et al., 2017; Mehta et al., 2017). However, human poses are highly dependent in time, and the sequential properties may be very useful to recover 3D poses and actions.

Thus, in this paper, we propose a novel method for recovering 3D human poses from images by using the sequential properties in 3D poses. For this objective, we combine the standard convolutional neural network (CNN) with Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997). The LSTM can represent the sequential properties in 3D human poses, and hence our network can recover 3D human pose at each time instant considering the sequence of human motions. As as result, our method can recover 3D human poses, even if some body portions are occluded by other body portions.

Furthermore, our network considers not only 3D human poses, but also 3D motions of a camera which observes the human. For separating 3D human moti-

Shimizu, T., Sakaue, F. and Sato, J.
Recovering 3D Human Poses and Camera Motions from Deep Sequence.
DOI: 10.5220/0006718603930398
In *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2018) - Volume 5: VISAPP*, pages 393-398
ISBN: 978-989-758-290-5

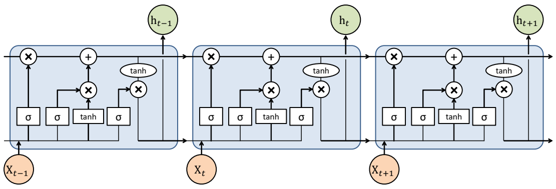


Figure 1: The structure and the state transition of LSTM. X_t and h_t denote input and output at time t. σ denotes a sigmoid function, which acts as a gate of data flow. By controlling these gates, the LSTM can preserve sequential information and learn time varying properties.

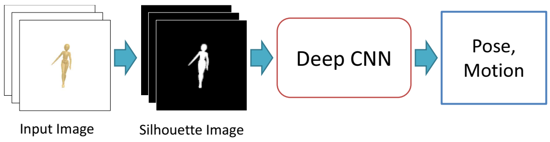


Figure 2: The outline of human pose and camera motion estimation.

ons and 3D camera motions, we fix the basis 3D coordinates at the waist of a human body, and 3D human motions and 3D camera motions are described based on this basis coordinates. By using our method, we can estimate 3D human poses and 3D camera motions simultaneously.

In section 2, we briefly review the convolutional neural network (CNN) and the Long Short-Term Memory (LSTM). In section 3, we propose a method for estimating 3D human poses and camera motions by combining CNN with LSTM. The results from the proposed method are shown in section 4, and the conclusions are described in the final section.

2 CNN AND LSTM

While the fully connected neural network learn the weight of connection between individual nodes in adjacent layers, the convolutional neural network (CNN) consists of convolution layers which connect adjacent layers by convolution, and learns the network by optimizing the kernels of convolution. As a result, CNN can optimize feature extraction from images, which had been conducted by man made feature detectors such as SIFT and HOG traditionally. Nowadays, CNN is the world standard in image recognition and used in various applications.

Although CNN is very useful and efficient in image recognition, the output of CNN is determined just from the current input images. As a result, it cannot process sequential data such as movies properly, since the output of sequential data depends not only on the current input, but also on the past input data.

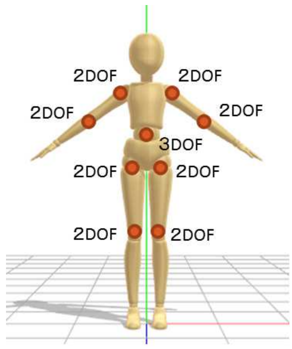


Figure 3: 3D human body model and the DOF of each joint.

For learning sequential data, Recurrent Neural Network (RNN) has been proposed (Mikolov et al., 2010).The recurrent neural network preserves sequential past data as the internal state, and can process sequential data properly. For learning long term dependency in sequential data, Long Short-Term Memory (LSTM) has also been proposed (Hochreiter and Schmidhuber, 1997). While the original RNN can only process short term data, LSTM can learn long term properties of data.

Fig. 1 show the network structure and the state transition of LSTM. The LSTM controls learning process by using gates, σ. The input gate controls input from the previous time, and the output gate controls the effect of the current layer to the next layer. The forget gate controls the destruction of data which are no longer needed. By controlling these gates, the LSTM can preserve sequential information and learn time varying properties in the data efficiently.

3 HUMAN POSE AND CAMERA MOTION ESTIMATION FROM CNN AND LSTM

In this research, we combine CNN and LSTM for estimating 3D human poses and camera motions simultaneously. For avoiding the effect of the variation of background scenes, we first transform camera images into silhouette images of human body and use the silhouette images as the input of our network as shown in Fig. 2.

3.1 Representation of 3D Human Poses

In this research 3D human poses are represented by a set of rotation angles at body joints. Suppose we have N joints in a human body. Then, since each joint has 3 rotation axes, the human pose can be represented by $3N$ rotation parameters.

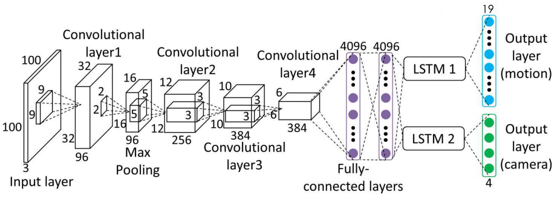


Figure 4: Proposed network structure for 3D human pose and camera motion estimation.

However, the rotations of arms and legs around their axes are irrelevant to human poses. Therefore, we consider each of the upper arms, lower arms, upper legs and lower legs has only 2 DOF, and only the waist has 3 DOF. Thus, in this research, we represent the 3D human pose by using 19 parameters as shown in Fig. 3.

The world coordinates are fixed at the waist of the human body, and 3D positions and orientations of all the objects in the scene are represented by using the waist based world coordinates.

3.2 Representation of Camera Motions

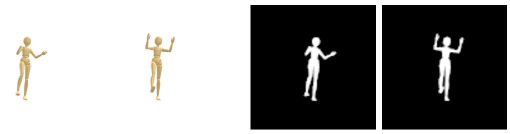
In this research, we assume that not only the human body but also the camera which observes the human body moves in the sequential observations. Thus, we estimate camera positions as well as human poses at each time instant. We assume that the viewing direction of the camera is fixed to the center of the human body, i.e. waist, and the camera positions are represented by using the orientation, θ, ϕ, and the distance d from the waist based world coordinates. The camera can also rotate around the viewing axis with ω. Thus, the camera position and orientation have 4 parameters.

In this research we estimate these 4 parameters of camera motions as well as 19 parameters of human poses. Hence, we estimate totally 23 parameters.

3.3 Network Structure

We next describe the network structure of the proposed method. In this research, we combine CNN and LSTM for estimating 3D human poses and camera motions simultaneously by using the sequential properties of human motions and camera motions efficiently.

Suppose we have an input image $\mathbf{x}_t$ from the camera at time t. Then our network estimates camera motion parameters $\mathbf{C}_t$ and human pose parameters $\mathbf{P}_t$ at time t from the input image $\mathbf{x}_t$. Considering the sequential properties of human poses and camera motions, our network can be considered as a function F which estimate the current state of the network $\mathbf{S}_t$ as well as the camera parameters $\mathbf{C}_t$ and human pose parameters $\mathbf{P}_t$ from the current input image $\mathbf{x}_t$ and the previous state $\mathbf{S}_{t-1}$ of the network as follows:



(a) input images (b) silhouette images

Figure 5: Examples of input images and silhouette images.

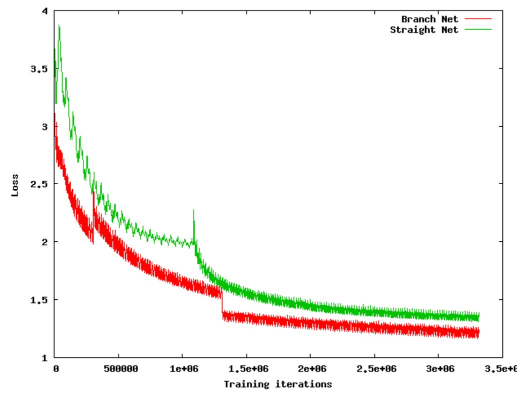


Figure 6: Changes in test loss in network training. The red line shows the loss of the proposed branch net which uses 2 separate LSTMs for human pose and camera motion, and the green line shows the loss of a straight net which uses a single LSTM for both human pose and camera motion.

$$\{\mathbf{C}_t, \mathbf{P}_t, \mathbf{S}_t\} = F(\mathbf{x}_t, \mathbf{S}_{t-1}) \tag{1}$$

Thus, learning of the network is considered as the estimation of function F by regression analysis.

For realizing the estimation, our network consists of 4 convolution layers, a pooling layer and 2 fully connected layers followed by 2 sets of LSTMs and fully connected layers as shown in Fig. 4. Our network first extract image features by using 4 convolution layers and a pooling layer. Then 2 fully connected layers transform the result into a low dimensional feature vector. Then, the result is separated and analyzed by two different LSTMs, one for the estimation of human pose parameters and the other for the estimation of camera motion parameters. These LSTMs derive feature parameters of human pose and camera motions updating their internal state. Then, the final layers transform these feature parameters into 19 human pose parameters and 4 camera motion parameters.

In this network, we consider the transition of human pose and the transition of camera position are independent to each other, and estimate the human poses and camera motions by using 2 different LSTMs. By using the LSTM, we can estimate 3D human po-

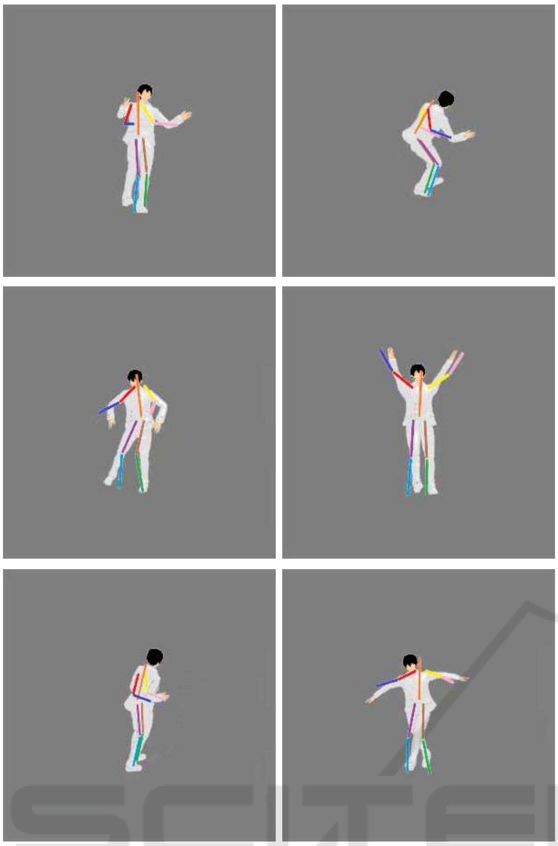


Figure 7: The result of 3D human pose estimation. The estimated 3D human poses were reprojected into the original images by using the estimated camera motion.

ses efficiently, even if some body portions are occluded by other body portions, which happens often in silhouette images. By learning the network from the back propagation, we realize the simultaneous estimation of human poses and camera motions.

3.4 Learning Network by using CG Models

We next consider the training of our network. For training the network avoiding overlearning, we need huge amount of training data in general. However, it is not easy to obtain huge amount of image data of human poses under various camera motions in the real scene. Therefore, we in this research use synthetic images generated by using CG models.

We generated human models with various body shapes, and added various pose parameters to them. We also generated a virtual camera with various motions, and observed the human poses to generate sequential CG images. For generating the pose of human, we used Mocap database (Lab, 2003) provided by Carnegie Mellon University. The Mocap database consists of 2605 different motions, such as walking, dancing, playing sports etc. We used 2000 of them for training and used 605 of them for testing in the synthetic image experiments. The virtual camera was moved around the human body fixating the viewing direction to the center of the world coordinates, i.e. center of the waist of the human body.

The use of synthetic images enables us to learn large variations of human pose parameters and camera motion parameters easily and efficiently. We can also simulate various types of human body, and control these parameters according to the objective of application systems. By using the synthetic training data, we train our network efficiently, and use it for estimating human poses and camera motions simultaneously.

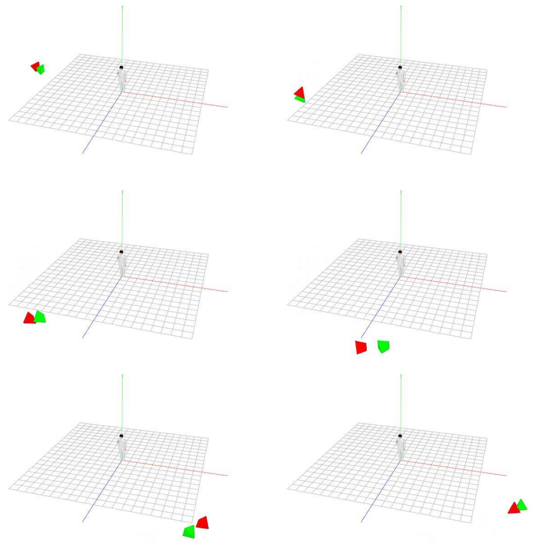


Figure 8: The result of camera motion estimation. The red quadrangular pyramid shows the estimated camera positions and orientations, and the green quadrangular pyramid shows the ground truth.

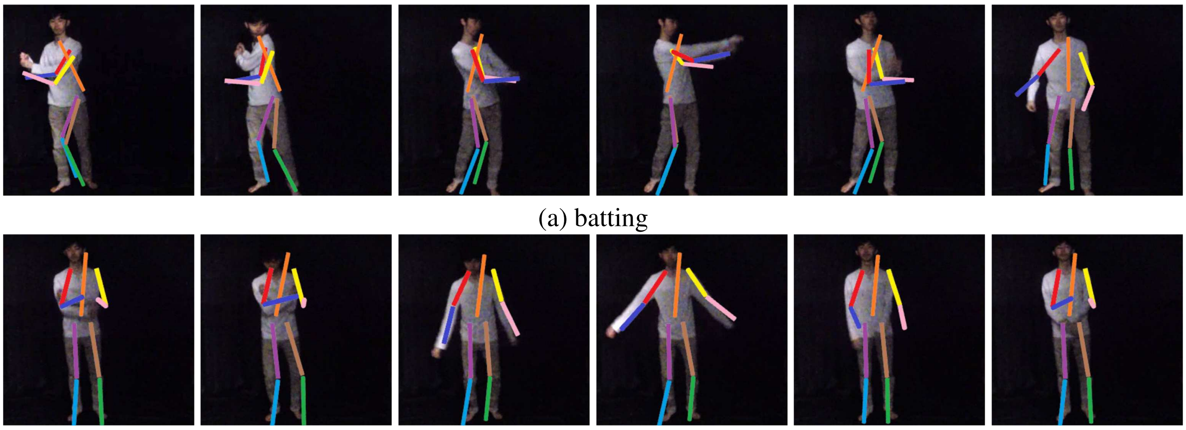
Table 1: The error of 3D human pose estimation and camera motion estimation with and without LSTM.

	human pose (°)	camera (m)
with LSTM	11.8	2.6
without LSTM	18.2	5.7

4 EXPERIMENTS

We next show the results of simultaneous estimation of human poses and camera motions by using the proposed network. The experiments are conducted by using synthetic images as well as real images.

In our experiments, a 3D human body model



(a) batting

(b) exercise

Figure 9: The result of 3D human pose estimation. The estimated 3D human poses were reprojected into the original images by using the estimated camera motions.

shown in Fig. 3 was used for synthesizing training images. As we explained in section 3.4, we used Mocap database (Lab, 2003) for generating 3D human poses. The synthetic images were generated by changing human poses and camera motions. The image size was 100×100. Fig. 5 shows some examples of synthetic images and their silhouette images. We generated 9000 sets of 10 sequential images from 2000 motions in Mocap database randomly, and used them for training our network. We also generated 1200 sets of 10 sequential images from the remaining 605 motions in Mocap database randomly, and used them for testing in the synthetic image experiment. The network training was executed by using Caffe framework.

We first evaluated the efficiency of our network structure, which uses 2 different LSTMs for human pose estimation and camera motion estimation. For comparison, we also evaluated a network which estimates human poses and camera motions by using a single LSTM at the middle of our network shown in Fig. 4. Fig. 6 shows the changes in test loss in these 2 networks. The red line shows the loss of the proposed branch net which uses 2 separate LSTMs for pose estimation and camera motion estimation, and the green line shows the loss of a straight net which uses a single LSTM for both pose estimation and camera motion estimation. As shown in this figure, the test loss of the proposed network decreases much faster than that of the straight net. This is because the proposed network can learn the pose and motion parameters more efficiently without learning irrelevant parameters by separating pose net and motion net.

We next show the results of 3D human pose estimation from synthetic images in Fig. 7. The estimated 3D poses were reprojected into the original input images by using the estimated camera motions in this figure. As shown in these images, various poses were estimated well by using the proposed network. Fig. 8 shows the camera motions estimated by the proposed network. The red quadrangular pyramid shows the estimated camera positions and orientations, and the green quadrangular pyramid shows the ground truth. As shown in this figure, the 3D camera motions were also estimated properly. The accuracy of estimated 3D human poses and 3D camera positions is as shown in table 1. For comparison, we also evaluated the accuracy of a network without LSTM. As shown in this table, the proposed network with LSTM provides us with much better accuracy, and we find that the use of sequential properties of pose and motion is very important.

Finally, we show the results of 3D human pose estimation from real image sequences. Fig. 9 shows sequential images of batting motion and exercise motion, and the estimated 3D human poses projected into images. The silhouette images were extracted by using the background subtraction method in these experiments. Although there are some estimation errors in the output of our network, the estimated results are reasonable.

These results show that the proposed method enables us to estimate sequential 3D human poses and camera motions properly.

5 CONCLUSION

In this paper, we proposed a novel method for recovering 3D human poses and camera motions from sequential images by using CNN and LSTM. While the existing methods recover just 3D poses relative to the viewpoints, our method estimates 3D human poses and 3D camera motions simultaneously. For using the sequential properties of human poses and camera motions, we combined CNN with LSTM, and showed that they can represent sequential properties in input data properly. We also showed that the network structure which uses 2 separate LSTMs for 3D pose estimation and camera motion estimation is efficient.

REFERENCES

Agarwal, A. and Triggs, B. (2004). 3d human pose from silhouettes by relevance vector regression. In *Proc. CVPR*.

Chen, C.-H. and Ramanan, D. (2017). 3d human pose estimation = 2d pose estimation + matching. In *Proc. CVPR*, pages 7035–7043.

Guan, P., Balan, A. W. A., and Black, M. (2009). Estimating human shape and pose from a single image. In *Proc. ICCV*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Lab, C. G. (2003). Mocap: Motion capture database. In *http://mocap.cs.cmu.edu/*.

Le, Q. V. (2013). Building high-level features using large scale unsupervised learning. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, pages 8595–8598.

Le, Q. V., Karpenko, A., Ngiam, J., and Ng, A. Y. (2011). Ica with reconstruction cost for efficient overcomplete feature learning. In *Advances in Neural Information Processing Systems*, pages 1017–1025.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lin, M., Lin, L., Liang, X., Wang, K., and Cheng, H. (2017). Recurrent 3d pose sequence machines. In *Proc. CVPR*.

Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H.-P., Xu, W., Casas, D., and Theobalt, C. (2017). Vnect: Real-time 3d human pose estimation with a single rgb camera. In *Proc. SIGGRAPH*.

Mikolov, T., Karafia, M., Burget, L., Cernocky, J., and Khudanpur, S. (2010). Recurrent neural network based language model. In *Proc. INTERSPEECH*.

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. (2011). Real-time human pose recognition in parts from single depth images. In *Proc. CVPR*.

Sminchisescu, C. and Telea, A. (2002). Human pose estimation from silhouettes : a consistent approach using distance level sets. In *Proc. International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision.*

Taylor, G., Fergus, R., LeCun, Y., and Bregler, C. (2010). Convolutional learning of spatio-temporal features. *Proc. ECCV*, pages 140–153.

Tome, D., Russell, C., and Agapito, L. (2017). Lifting from the deep: Convolutional 3d pose estimation from a single image. In *Proc. CVPR*.

Toshev, A. and Szegedy, C. (2014). Deeppose: Human pose estimation via deep neural networks. In *Proc. CVPR*, pages 1653–1660.